Towards Good Web Conferencing System Design for Ubi-Media Collaboration

Bin Yu[1], Yong Rui
Microsoft Research
One Microsoft Way, Redmond, WA 98052, U.S.A.
{t-binyu, yongrui}@microsoft.com

ABSTRACT

Media, data and documents are becoming ubiquitous in the past decade, across time zones and across organization boundaries. Web conferencing systems emerged as effective tools for people to communicate and collaborate. These systems combine multiple distributed applications to manipulate multimedia content within a conferencing session, and achieved great results. However, despite great development in such systems, current products on the market are still far from perfect in many aspects. In this paper, we first present an overview of existing systems, and then identify good system design principles based on our own experience with the existing systems. Finally, we outline several interesting research directions that may further enhance users' collaboration experience.

1. Introduction

Web conferencing systems (WCSs) are the latest and one of the most promising developments in the collaboration world. Its primary applications include project/team collaboration, training, marketing presentation, and customer support services. We define a WCS as a combination of hardware and software utilities that enable information exchange between conferencing participants via computer networks in real time. The hardware utilities include computing devices, input/output peripherals and communication networks. Example software utilities include presentations, web co-browsing, whiteboard sharing, application/desktop sharing, file transfer, text chatting, instant messaging, audio/video communication, polling, and session recording/playback. Figure 1 is a screenshot of Centra [2], a WCS on the market, from its user manual.

Various conferencing systems, e.g., [1][2][5][6][7][8][9][10][19], have been on the market for many years. But their importance was not fully appreciated until the down cycle of the U.S. economy in 2000 and the tragic event of 9-11. As more users had gone through the learning curve, they started to realize the convenience and power of WCSs. Specifically, WCSs are easier to set up and normally cheaper than conventional video conferencing system [13]. In addition, WCSs do not require dedicated ISDN lines. Compared with audio conferencing systems, WCSs allow people to see each other, which is especially important during the initial meetings. Furthermore, WCSs are better than both video and audio conferencing in that they support easy data collaboration, e.g., application sharing and presentation viewing. According to Frost & Sullivan [7], the worldwide WCS market stood at $472.1 million in 2003, and is expected to reach $3.02 billion by 2010.

Note that WCSs are different from conventional Instant Messaging (IM) systems, such as Yahoo Messenger [20], in that IM systems primarily facilitate people to exchange quick textual messages, while WCSs provide much more powerful visual/audio channels for collaborative discussion on more complicated problems. However, one interesting trend in the WCSs development is to invoke collaboration tools (such as PowerPoint presentation, audio/video conferencing and whiteboard) from within the IM interface.

There exist several comprehensive surveys and guides on web conferencing products and services from end users' perspective, e.g. [3][14][15]. There is, however, little research on WCS design issues. This paper tries to fill that gap. Specifically, we want to

1. identify WCS design principles that should be encouraged or avoided, based on the authors' first-hand experiences; and
2. suggest future research directions that may lead to significant improvement in user experience with WCSs.

The rest of this paper is organized as follows. We will first give an overview of existing WCSs in Section 2, and then present system design principles in Section 3. We promote several future research directions in Section 4, and conclude the paper in Section 5.

2. Overview of Existing WCSs

There are many WCSs on the market today. In general, there are two major competing trends in the WCS industry. On one hand, some systems target at large organizations to reduce their huge travel costs by providing comprehensive solutions. Example systems include WebEx [19] and Centra [2]. Such systems generally provide more sophisticated

[1] This project was done during the first author's internship at Microsoft Research. His current email is binyu@uiuc.edu. To provide an unbiased view of existing WCSs, because of authors' affiliation, we deliberately do not discuss Microsoft Live Meeting in this paper.

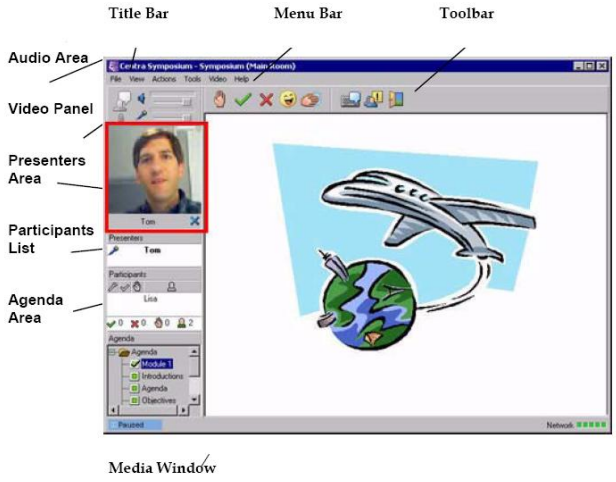

Figure 1. An example WCS web interface from Centra [2].

facilities, e.g., better audio-visual quality, and more software utilities. This is at the expense of higher price and higher resource requirement in terms of network bandwidth, computation power and capturing devices.

On the other hand, some WCSs weight simplicity with more importance, and they focus on a few key features that work well enough for most daily activities. Representative simple WCSs include Glance [8], Convoq ASAP [5], Sightspeed [17] and Linktivity Webdemo[10]. They focus on smaller scale conferencing scenarios for individual users and small companies and often ask for much lower prices. For example, Glance [8] simplifies several collaboration features into the application sharing facility, which may not be best for large group of participants, but fits well to the needs of smaller group of users.

Despite the differences in comprehensive and simple systems, there are many common issues that both camps need to properly address. In the rest of the paper, we want to identify such design principles that are worth to follow for all WCSs, and promising future directions that may eventually lead to a solution that best serves the customers.

3. Detailed System Design Principles

3.1 Ease of Use:

Ease of use has always been the top design principle for all computer systems, and web conferencing is no different. After using and studying many existing systems, we summarize the following key principles that can lead to ease of use.

3.1.1 Following Users' Previous Experience

The easiest way to teach a user how to use a new tool is to make it work like an old tool that the user is already familiar with. By previous experience, we mean both the experience in using other computer tools and the experience that are not related to computers at all. For example, the Glance [8] system achieves great simplicity by unifying almost all conferencing tasks into one facility – desktop sharing. For example, distant PowerPoint presentation is extremely easy because the presenter does nothing more than running a local PowerPoint presentation. Similarly, Convoq ASAP [5] is very easy to use for all users who are already familiar with IM tools such as Yahoo Messenger [20], because all conferencing facilities are naturally built on top of existing IM interface.

This principle of "experience-driven design" can be used to simplify many tasks. For example, the "file sharing" feature is not straightforward in most WCSs because a user often has to explicitly "send", "receive", "upload" or "download" the files. However, considering that users are familiar with simple "copy" and "paste" or "drag and drop" operations, we can create a pseudo drive/directory on the local PC's file system that represents remote peers so that users can resort to what they already know.

3.1.2 Hide What Is Not Being Used

Many a time the complexity of using a WCS comes when a user cannot easily find the controls that s/he needs, because there are too many menus and buttons. For example, as the authors tried out some WCSs that provide rich functionalities, we found it's not easy to find the menu item or button on the toolbar when we wanted to quickly change or check some settings or repeat the same action. To solve this issue, it is better to define several working "modes", e.g., "application sharing mode", "Internet co-browsing mode", "presentation mode", so that in each mode the toolbar and menu items are adapted to the hot tasks for that mode.

3.2 Customization

Customization gives a user the control over what kind of service he/she enjoys, and we believe a good WCS should provide customizability in the following aspects:

3.2.1 Resource Consumption Level

Different users have different resource availabilities in terms of hardware capacity (CPU, memory, disk, screen space, etc.), and network connection bandwidth. Even if two users have similar availabilities of resources, they may still want to allocate different amount of resource to a particular conferencing session. Therefore, it is desirable that a WCS gives advanced users the choice of specifying how much resource the system can consume. For example, in terms of screen space for the user interface, it may not be a good idea to include all the functions into one large window that takes the entire desktop, not to mention in an input-blocking mode, because the user may well want to take care of other applications at the same time. NetMeeting [11] has done a good job in this aspect by using a small window for each task, and left users the control of allocating the screen space.

Along this line of reasoning, all types of Quality of Service (QoS) should be customizable based on resource availability and user requirements, such as video quality, audio quality, PowerPoint presentation quality and session recording quality. From our observation, very few WCSs provide option for users to specify what the resource budget is and what level of quality is desired. With the dynamic transcoding/transformation techniques, such adaptations are realistic to implement.

3.2.2 Implementation Polymorphism

Polymorphism refers to function overloading in object oriented programming languages, i.e., for the same method call, different objects act differently based on their own purpose of existence. We can borrow this idea into the design of WCSs because different users may have different computing devices or resource availabilities, and we want the same function to be carried out in different ways based on each individual situation.

For example, for the PowerPoint presentation facility, currently almost all existing WCSs convert the PowerPoint file into an image format, e.g., PNG or JPEG, and then present the same presentation to all viewers. This "one size fits all" approach may not be adequate: some users may have high bandwidth connection and PowerPoint installed on their machine, so that they can watch a PowerPoint presentation that runs on their own local machines and is controlled by the remote presenter; on the other hand, some users may only have a small iPAQ with limited wireless connection, and the JPEG images would not fit into the screen. Here, if we apply the idea of polymorphism, we can allow the same PowerPoint presentation function to be realized in a customized way for each participant:

- For those clients with PowerPoint application installed, whether it is on Windows, Linux or Macintosh platform, download the PowerPoint file and then run PowerPoint locally. The presenter's interactions (mouse movement or keyboard events) can be sent to the participant's local PC to follow what would happen at the presenter's PC.
- For those users without PowerPoint application installed, they are sent differently sized JPEG images based on how much bandwidth they have and how large their screen sizes are.

Note that this customizability may seem contradictory to the principle of "ease of use". But in fact they are talking about two different dimensions. Customizability refers to the process of setting up the WCS, while ease of use refers to the process of using the WCS. Of course, the customization step itself can be made easy to use, too. In addition, ease of use is more desired for novice users, and as they become more familiar and experienced with the system, customization will become important for them. Based on our observation, it also a good idea to for WCSs to provide several sets of predefined composition of the functionalities and parameter settings based on the typical user scenarios.

3.3 Reliability

No one wants to use a system that may crash during an important meeting. Because a WCS is in fact a composite session of several distributed applications, they are more vulnerable than ordinary systems. In such a situation, system designers need to take the following two principles into consideration:

3.3.1 Independent Service Component

When a fault occurs, it normally comes from one particular function. Therefore, if different facilities are as loosely coupled as possible, they may still work well when one or more facilities fail. For example, in many existing systems, one task often freezes the whole interface until it is done, such as PowerPoint file uploading or application sharing. That is, when a user is using these features, he cannot access any other features from the interface. This is very likely to cause the whole interface to crash if an error occurs during the file uploading. We believe that a more robust design is to model all the facilities as independent applications sharing a minimal set of session information.

3.3.2 Distributed Architecture

A centralized server is usually the source of problem because it has high traffic/computation pressure and it is often the target of attack. To increase scalability and robustness, a good WCS should be designed with distribution of functionality in mind. As personal computers become more and more powerful, more tasks could be off-loaded to the client end from the shared server. For example, currently most systems store the meeting content in a centralized server and then distribute to all participants on request. This is not reliable because if the content server is down, the meeting can not continue. Instead, considering the low price of hard disks, it is better to replicate the content to a selected set of powerful participant PCs before hand. In fact, some systems, e.g.,Groove [9] and Cybernar [6], are completely server-less and they adopt the Peer-to-Peer architecture to remove the single-point failure and utilize the computing power residing on each participating PCs. This is a very promising direction to go that reduces overall cost and increases reliability.

3.4 Security

Needless to say, security is very important for WCSs, and almost all the existing systems have made great efforts in securing the integrity and confidentiality of conferencing content. Two most commonly used protections are password authentication and 128-bit Secure Socket Layer **(**SSL) encryption. Because of page limitation, for a detailed discussion, readers are referred to [3].

4. Future Directions

Although quite many WCSs are on the market today, users' experience with them is still far from ideal. In this section, we propose a few interesting directions that may bring user experience to a higher level.

4.1 Asynchronous Ad-hoc Collaboration

Existing WCSs all focus on real-time collaboration. However, considering the popularity of the two widely used applications of the Internet, *www browsing* and *email*, asynchronous collaboration is a very important part of the collaboration experience. For example, sometimes it may be very hard to find a common time slot for a group of people to have a meeting together, and in many occasions people need to meditate and search for related materials before they raise their opinions in a group discussion. Consider a web forum way of asynchronous collaboration: one participant may start a discussion thread by sharing a piece of video and PowerPoint slides on the conferencing catalog, and then he gets offline. Later, when any other participant becomes online, he can review the existing posts, and take his time to check related information and organize his thoughts. Then he may add his comments into the discussion thread by adding new slides, notes, whiteboard drawings, a few web links, or even a piece of audio/video recordings. The updates to the thread will trigger alert notification to all interested parties who may be offline temporarily. When a brilliant idea suddenly strikes someone, he does not have to call for an emergency meeting, but only needs to "join" the asynchronous meeting and presents his thoughts right away.

4.2 Smart Recording and Playback

Recording and playback of web conferences is still an under-studied field. We propose four directions where improved user experience may be achieved.

- Event-based Recording/Playback

Almost all existing systems use *video*-based recording of what the on-site participants have seen and heard. The file size is normally very large, yet the video/audio quality is generally not satisfactory. Considering the fact that the state of the recording session evolves at each user interaction event, a past conference session can be completely reconstructed by simulating the original participants' interactions (*events*). Specifically, all the data files (e.g., PowerPoint slides, word documents, etc.) and all the user interactions that change the state of the conference session (e.g., document navigation, annotation, whiteboard strokes, session join/leave activities, etc.) are stored separately. Later on, when a user wants to playback the session, he can use the recorded events as natural indexes into the session, and jump to any event to start the playback. The system can quickly reconstruct the session state at the starting point as what it was during the original session, then all the following interactions are replayed to simulate what happened in the original session. There are several benefits of *event*-based recording/playback: 1) a natural timeline with event-based indexes is created as the conference session evolves; 2). Playback quality is higher because the users see exactly what the original session would be, while for *video*-based recording there are normally quality loss due to down-sampling and compression. 3). If the user wants to download the recording for playback, he will only download the documents and interactions, so the file size can be greatly reduced compared to downloading a *video*-based recording. 4). For on-demand browsing of meetings, it requires much less bandwidth than streaming the actual video.

With the *event*-based recording/playback, several other improvements can be achieved

- Conference Session Retrieval

When the amount of recordings becomes large, users may desire to search through recordings of conferences for specific information. With traditional video/audio recording, because the video/audio content is not easily understood by computers, retrieval results are often dissatisfactory. However, with the *event*-based recording, two advantages exist: 1). It preserves the related documents in their original formats, which are normally textual, so we can leverage the more mature techniques in text-based retrieval. 2) Metadata, e.g., events, is logged, which can be used for retrieval.

- Conference Session Summarization

Summarization of a long conference session into a short highlight segment is often desirable for faster browsing and searching for specific information. With traditional video/audio recording, summarization is not easy because of the limitation in video/audio content understanding technologies. However, with event-based recording, we have the natural indexes that segment the session into small "shots", so summarization can be done by prioritizing these shots so that more interesting and important shots are presented to users who want a quick review of the session. Specifically, the priorities can come from two sources: content analysis and user interaction analysis.

For content analysis approach, the shots in a session can be weighted differently based on whether important events happened during that shot and whether the related document segments contain important/interesting information. Note that heuristics may be needed to determine the importance of events and what information is more important or interesting, and this knowledge may come from the particular conference topic or be manually specified by users.

For user interaction analysis approach, the log analysis approach [21] can be applied to utilize users' engagement and interactions to gain understanding of the session

semantics. Specifically, there are natural inherent "links" between media content such as PowerPoint slides, web slides, whiteboards and shared applications, and "interesting" pieces of content will tend to be strongly correlated to each other. On the other hand, playback users differ in their interests in the materials and engagement in the browsing process, so their interaction will have different weights in determining the value of the content segments they visit. Therefore, by utilizing the reinforcement between engaged users and interesting content pieces, we can rank the recorded content pieces in their interestingness to users and assist future users in their browsing.

- Adaptable Recording

Currently almost all WCSs "publish" final recording files that can not be updated later. With event-based recording, the whole session is segmented into small shots based on different events, which provides the opportunity for later users to "re-join" a past conference session by inserting his own interactions. For example, for a distant learning class, the teacher may ask each student to present his/her ideas on a topic. Imagine that one student is absent from that class, he can make up the class by browsing through the recording, and insert at the middle of the conference session his own responses. And for later reviewers, the last student's interactions are presented just like other students, though some notation may be added to indicate this is a make-up recording segment.

4.3 Anonymity

In many scenarios, e.g., sales presentation, participants of a conference are strangers to each other. Though they will authenticate themselves, they may not want others (sometimes even the presenter) to know that they are in the session. Some users may want to hide themselves initially, and then "show up" when they have some questions. As far as we observe, no existing system deals with this requirement.

4.4 Improved Video Conferencing Capability

Because of the high bandwidth requirement for the video channel, most existing WCSs take video as an add-on component that is normally not activated. However, face-to-face contact has always been a very important factor in human communication, as facial expressions and body languages often tell more than words. In addition, as pointed out in a recent survey conducted by RoperRAW [18], people tend to be more focused in a face-to-face meeting, while they may do other work such as checking emails or surfing the Internet if no video image is being captured. To increase meeting/marketing effectiveness, providing better video quality is important for next generation WCSs. As the last mile bandwidth grows higher and as new video coding standards such as H.264 and MPEG4 become widely used, video will become more accessible in the future.

4.5 Beyond What We Have Today

A web conferencing session is more than a composite of several distributed applications for the following two reasons: (a) there can be multiple users giving inputs to the same application, and (b) these applications run at the same time and share common session metadata and critical resources, such as CPU power and screen space.

There could be many new application scenarios that were not thought about before because of (a). For example, before the wide use of WCSs, it is rare that multiple users will work on the same document at the same time. However, now that every session participant has a set of mouse and keyboard, they can work on the same whiteboard, PowerPoint slide, word document or spreadsheet at the same time. Imagine that if a chip design company has an urgent project that normally takes each designer three days to finish, now they can have these three designers work together on the same blueprint at the same time in parallel and finish the work in one day.

On the other hand, because of (b), applications that work perfectly alone may need additional functions when used closely with other applications. For example, currently PowerPoint presentations take full screen mode only, but when used in a web conferencing session, it may be desired that it can run in a particular window, such that it allows overlay of a small Picture-in-Picture window that shows a video window or another PowerPoint slide or even a web page.

5. Conclusion

WCS is a key solution to improving people's ubiquitous access to media, data and information. As a result, it can significantly improve people's efficiency and productivity in their daily business, especially when they are distributed. Based on our own research experience as well as the observations on commercially available WCSs, we reviewed the strength and weakness of existing WCSs, proposed a set of design guidelines, and discussed interesting future research directions. As the world is becoming more flat, we envision that future WCSs will play more and more important roles in helping people access, manage and collaborate on their data.

6. Reference

[1] Best Web And Videoconferencing Alternatives To WebEx, Live Meeting, Centra For The Small, Non-Profit And Home-Based Company, http://www.masternewmedia.org/2003/11/13/best_web_and_videoconferencing_alternatives.htm

[2] Centra, http://www.centra.com/

[3] Cisco, Best practices in web conferencing security, http://itpapers.techrepublic.com/abstract.aspx?docid=83438&promo=400111&lop=wpr.20,1011, white paper

[4] Conferguide 2004, web conferencing guide, http://www.conferzone.com/services/conferguide_thoit.html

[5] Convoq ASAP, http://www.convoq.com/

[6] Cybernar from Terra Firma, http://www.terrafirmacomm.com/products/cybernar.ph

[7] Frost and Sullivan, http://www.frost.com/

[8] Glance, http://www.glance.net/

[9] Groove, http://www.groove.net/

[10] Linktivity webdemo, http://www.linktivity.com/

[11] NetMeeting, http://www.microsoft.com/windows/netmeeting/

[12] Orbitalk, http://www.orbitalk.com

[13] PolyCom, http://www.polycom.com

[14] Rich Media Conferencing 2004, http://www.wainhouse.com/reports/rmc2004.html

[15] Robin Good's official guide to web conferencing and live presentation tools, http://www.masternewmedia.org/reports/webconferencing/guide/

[16] Rui, Y., Huang, T. S., and Mehrotra, S. , "Constructing table-of-content for videos", Multimedia Systems, V 7, Issue 5, September 1999

[17] Sightspeed, http://www.sightspeed.com

[18] Wainhouse Research Bulletin Archives, Vol. 4, Issue #44, http://www.wainhouse.com/files/wrb-04/WRB-0444.pdf, Dec 8, 2003

[19] WebEx, http://www.webex.com

[20] Yahoo Messenger, http://messenger.yahoo.com/

[21] Yu, B., Nahrstedt, K., Video Summarization based on User Log Enhanced Link Analysis, ACM Multimedia 2003